Check, Locate, Rectify: A Training-Free Layout Calibration System for Text-to-Image Generation

Biao Gong[1†✉], Siteng Huang[2†*], Yutong Feng[1], Shiwei Zhang[1], Yuyuan Li[2], Yu Liu[1]
[1]Alibaba Group [2]Zhejiang University
{a.biao.gong, siteng.huang}@gmail.com y2li@zju.edu.cn
{fengyutong.fyt, zhangjin.zsw, ly103369}@alibaba-inc.com

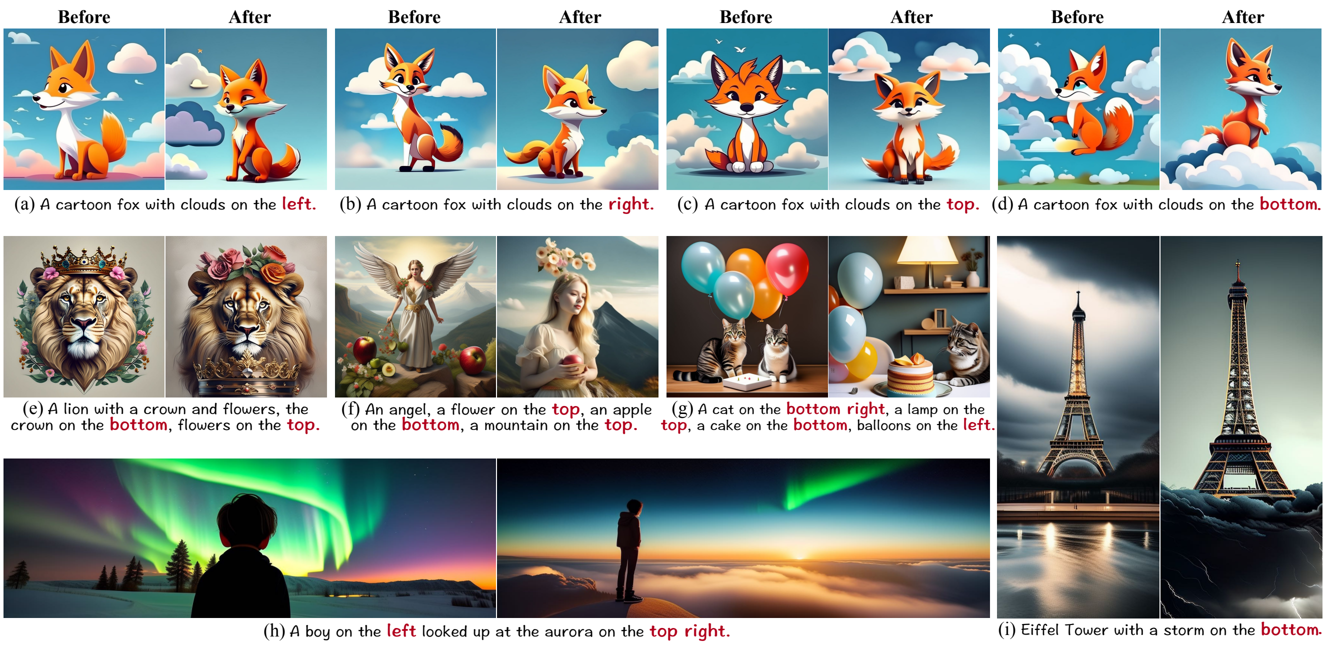

Figure 1. Given only the input textual prompt, our system can autonomously detect and rectify the layout inconsistencies across various **position requirements** (a-d), **object quantities** (e-g), and **resolutions** (h-i).

Abstract

Diffusion models have recently achieved remarkable progress in generating realistic images. However, challenges remain in accurately understanding and synthesizing the layout requirements in the textual prompts. To align the generated image with layout instructions, we present a training-free layout calibration system `SimM` that intervenes in the generative process on the fly during inference time. Specifically, following a "check-locate-rectify" pipeline, the system first analyses the prompt to generate the target layout and compares it with the intermediate outputs to automatically detect errors. Then, by moving the located activations and making intra- and inter-map adjustments, the rectification process can be performed with negligible computational overhead. To evaluate `SimM` over a range of layout requirements, we present a benchmark SimMBench that compensates for the lack of superlative spatial relations in existing datasets. And both quantitative and qualitative results demonstrate the effectiveness of the proposed `SimM` in calibrating the layout inconsistencies. Our project page is at `https://simm-t2i.github.io/SimM`.

1. Introduction

Text-to-image generation [11, 20, 29, 31] has emerged as a promising application of AI-generated content (AIGC), demonstrating the remarkable ability to generate synthetic images from conditional text descriptions. This technology has attracted considerable attention in recent years due to

†Equal contribution. ✉ Corresponding author.
*Work done during internship at Alibaba Group.

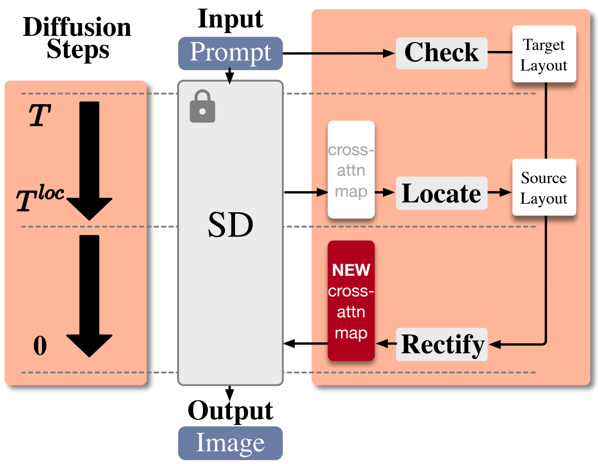

Figure 2. **The "check-locate-rectify" pipeline of `SimM`**, intervening in the generative process on the fly during inference.

its potential impact on various domains such as image customization [34, 45], 3D content creation [22, 26] and virtual reality [4]. Since achieving high-quality and diverse image generation is challenging, recent advancements have witnessed the rise of diffusion models [16, 32]. Diffusion models employ a sequential generation process that gradually refines the generated images by iteratively conditioning on noise variables. This iterative refinement mechanism allows for an improvement in the fidelity and quality.

Despite the effectiveness of diffusion models, a significant challenge remains: most text-to-image generators, typified by Stable Diffusion [32], show limitations in accurately understanding and interpreting textual layout instructions [12]. This can be regarded as a kind of "hallucination" [13, 48], which refers to the phenomenon that the generated image is inconsistent with the prompt content. On the one hand, various textual descriptions include the relative relation "*a dog to the left of a cat*" and the superlative relation "*the crown on the bottom*", presenting an inherent difficulty for automated systems to parse and understand layout information. Besides, inaccuracies in spatial relations may be due to the prior knowledge embedded in pre-trained models, as the large dataset may contains certain biases or assumptions about object placement or orientation. To exemplify this point, consider the following situation: since the "*crown*" in the training images are predominantly positioned over the head of another organism, it becomes difficult to specify their occurrence below (Fig. 1-e).

These factors not only compromise the quality and fidelity of the generated images but also hinder the overall utility and user experience of text-to-image generation systems. Some efforts [43, 47] attempt to address the issue by training auxiliary modules or fine-tuning diffusion models on datasets with layout annotations. Apart from the difficulty of collecting sufficient high-quality data, these resource-intensive methods require retraining for each given checkpoint, making them struggle to keep up with the rapid version iterations of base models.

In this paper, we delve into the exploration of layout calibration given a pre-trained text-to-image diffusion model. Consequently, we present a training-free real-time system `SimM`, which follows the proposed "*check-locate-rectify*" pipeline. The **checking** stage is first applied to mitigate the potential impact on the generation speed, where `SimM` generates approximate target layout for each object by parsing the prompt and applying heuristic rules. After comparing the target layout with the intermediate cross-attention maps, layout rectification can be initiated if there are layout inconsistencies, and `SimM` locates the misplaced objects during the **localization** stage. Finally, during the **rectification** stage, `SimM` transfers the located activations to the target regions, and further adjusts them with intra-/inter-map activation enhancement and suppression. The entire workflow only affects the generation process, avoiding any additional training or loss-based updates.

We conduct both quantitative and qualitative experiments to evaluate the effectiveness of the proposed `SimM`. Since the popular DrawBench dataset [35] only contains prompts with relative spatial relations, we present a new benchmark SimMBench that includes superlative descriptions composed of various orientations and objects, compensating for the diversity of textual prompts. Compared to the recent works [6, 25, 47], which rely on precise target layout provided by the user, `SimM` achieves satisfactory correction results even when the target layout is not precise enough, leading to a significant improvement in the layout fidelity of the generated images.

2. Methodology

In this paper, we aim to align the generated images with the layout requirements in the prompts, and present a layout calibration system that requires no additional fine-tuning. In Sec. 2.1, we first briefly review the publicly avaliable, state-of-the-art text-to-image generator, Stable Diffusion [32]. In Sec. 2.2, we introduce how to determine whether a layout correction should be initiated. And in Sec. 2.3, we detail the localization of activated regions on the merged cross-attention maps. Finally, in Sec. 2.4, we present how the system rectifies the cross-attention activations according to the localized patterns and the target locations. An overview of the pipeline is illustrated in Fig. 2.

2.1. Preliminaries

Stable Diffusion. Stable Diffusion (SD) [32] applies a hierarchical variational autoencoder (VAE) [19] to operate the diffusion process [16] in a low-dimensional latent space. Specifically, the VAE consisting of an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$ is trained with a reconstruction objective. The encoder $\mathcal{E}$ encodes the given image $\mathbf{x}$ into latent features $\mathbf{z}$,

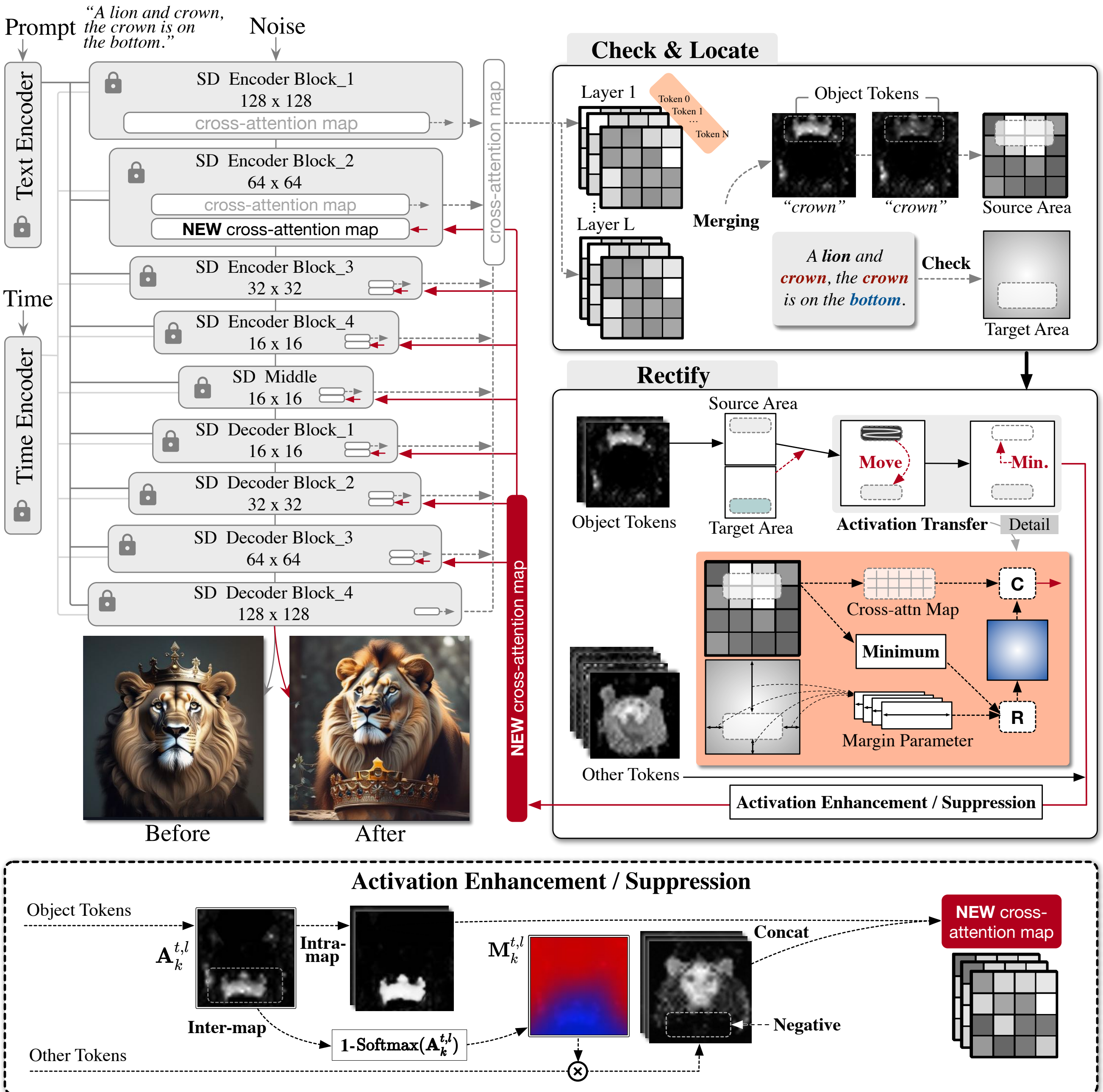

Figure 3. **A detailed illustration of our `SimM` system.** *R* means repeating, *C* means concatenating.

and the decoder $\mathcal{D}$ outputs the reconstructed image $\widehat{\mathbf{x}}$ from the latent, *i.e.*, $\widehat{\mathbf{x}} = \mathcal{D}(\mathbf{z}) = \mathcal{D}(\mathcal{E}(\mathbf{x}))$. To applied in a text-to-image scenario, a pre-trained CLIP [28] text encoder encodes the input textual prompt into N tokens $\mathbf{y}$, and a U-Net [33] consisting of convolution, self-attention, and L cross-attention layers is adopted as the denoiser ϵ_θ. During training, given a noised latent $\mathbf{z}^t$ and text tokens $\mathbf{y}$ at timestep t, the denoiser ϵ_θ is optimized to remove the noise ϵ added to the latent code $\mathbf{z}$:

$$\mathcal{L} = \mathbb{E}_{\mathbf{z}\sim\mathcal{E}(\mathbf{x}),\mathbf{y},\epsilon\sim\mathcal{N}(0,1),t}\left[\left\|\epsilon - \epsilon_\theta\left(\mathbf{z}^t, t, \mathbf{y}\right)\right\|_2^2\right]. \quad (1)$$

During inference, a latent $\mathbf{z}^T$ is sampled from the standard normal distribution $\mathcal{N}(0,1)$. At each denoising step $t \in [T, \cdots, 1]$, $\mathbf{z}^{t-1}$ is obtained by removing noise from $\mathbf{z}^t$ conditioned on the text tokens $\mathbf{y}$. After the final denoising step, the decoder $\mathcal{D}$ maps the latent $\mathbf{z}^0$ to an image $\widehat{\mathbf{x}}$.

Cross-Modal Attention. The SD model leverages cross-attention layers to incorporate textual cues for the control of the image generation process. Given the text tokens $\mathbf{y}$ and intermediate latent features $\mathbf{z}^l$, the cross-attention maps from the l-th layer $\mathbf{A}^l \in \mathbb{R}^{W^l \times H^l \times N}$ can be derived as

$$\mathbf{A}^l = \text{Softmax}\left(\frac{\mathbf{Q}^l \mathbf{K}^{l\top}}{\sqrt{d}}\right), \quad (2)$$

where $\mathbf{z}^l$ and $\mathbf{y}$ are projected to the query matrix $\mathbf{Q}$ and key matrix $\mathbf{K}$, the dimension d is used to normalize the softmax values, and we omit the superscript t for notational clarity and generality. Existing studies [5, 6] have proposed that for the object corresponding to the k-th token of the prompt, higher activations on the intermediate cross-attention maps $\mathbf{A}_k^l \in \mathbb{R}^{W^l \times H^l}$ indicate the approximate position where the object will appear. Therefore, we align the spatial location of generated objects with textual layout requirements by adjusting the activations on the cross-attention maps.

2.2. Check

A key constraint for the real-time system is to minimize the influence on the generation speed. Therefore, `SimM` first (1) detects the presence of object layout requirements within the text and (2) assesses any discrepancies between the generated image and the specified layout requirements. Only if both conditions are met does the system take corrective action; otherwise, it continues with normal generation to avoid additional computational overhead. The exact implementation of the two-step inspection is discussed below.

☑ **Layout requirements exist in textual prompts.** Existing studies [6, 47] have predominantly emphasized **relative** spatial relations that are more common in written language, such as "*a dog to the left of a cat*". However, we argue that **superlative** spatial relations, which refer to an object shares the same relation to all other objects, have been neglected by previous research and datasets [35]. For example, the phrase "*a flower on the left*" signifies that the flower is positioned to the left of all other objects, making it ideal for the leftmost target location. In practice, it is difficult for users to directly describe their layout requirements using multiple relative expressions at once, so more direct superlative expressions actually account for a larger number.

To effectively and efficiently capture both forms of expression in a straightforward manner, our system identifies specific positional keywords with predefined vocabulary (described in *Supplementary Material*). For **relative** spatial relations, we define five spatial relations, including *left*, *right*, *above*, *below* and *between*, with each relation containing a predefined vocabulary set. And for **superlative** spatial relations, we include additional vocabulary such as "*upper-left*" and "*lower-right*". The system filters out those prompts that contain words from the vocabulary set to determine the presence of layout requirements. In practice, such a simple check implementation achieves considerable accuracy with negligible additional computational overhead.

☑ **Discrepancy exists between the generated image and layout requirements.** To determine whether the generated image is consistent with the layout requirements, the target positions of all objects are necessary. For **target layout generation**, our system provides an efficient solution by performing a dependency parsing on the prompt following with heuristic rules. The dependency parsing can be implemented using an industrial-strength library such as spaCy [17]. After assigning syntactic dependency labels to tokens, `SimM` can parse the binary "`flower,leftmost`" from the superlative "*a flower on the left*", and the triple "`dog,left of,cat`" from the relative "*a dog to the left of a cat*". Following pre-defined rules, the system first assigns target boxes to objects associated with superlative position terms. Then, the remaining relative triples (and quaternions if "`between`" exists) can be organized as a semantic tree, with nodes as objects and edges as spatial relations. By traversing the tree, the remaining space in the image is successively allocated. A detailed example of assignment can be found in *Supplementary Material*. For the object of the k-th token, $\widehat{\mathbf{b}}_k = (\widehat{x}_k, \widehat{y}_k, \widehat{w}_k, \widehat{h}_k) \in [0,1]^4$ denotes the assigned bounding box, where $(\widehat{x}_k, \widehat{y}_k)$ is the relative coordinates of the centre, $\widehat{w}_k$ and $\widehat{h}_k$ are the relative width and height of the box. And the absolute boundaries $\widehat{\mathbf{b}}_k^l$ for the l-th layer can be computed with the concrete size of the corresponding attention map. Note that the predicted box may not necessarily fit the size of the object and is commonly larger. However, thanks to subsequent activation transfer, this does not affect the rectification performance.

Once the target boxes are obtained, the system prepares to assess whether each generated object is aligned with its target position. One natural solution, using an object detector on the generated image, requires a restart of the generation after the assessment for rectification and significantly increases the overall latency. Therefore, `SimM` places the alignment confirmation in the first denoising step (*i.e.,* the T-th step). Specifically, after deriving the cross-attention maps for all layers, a **layered attention merging** averages them to obtain a merged attention map:

$$\bar{\mathbf{A}}^T = \frac{1}{L}\sum_{l=1}^{L} \widetilde{\mathbf{A}}^{T,l}, \quad (3)$$

where $\widetilde{\cdot}$ means that the maps are first upsampled to a uniform resolution of $W^1 \times H^1$ before averaging. Then, for the object of the k-th token, `SimM` sums over the activations within $\bar{\mathbf{A}}_k^T$ that correspond to the bounding box $\widehat{\mathbf{b}}_k^1$. If the sum does not exceed a pre-defined threshold, the system predicts that the object will be generated in the wrong place.

2.3. Locate

After confirming the initiation of the rectification, the system identifies the source activated region for each object during the early T^{loc} denoising steps.

Temporal Attention Merging. For each time step $t \in [T, T - T^{loc}]$, the system simply saves the merged atten-

Figure 4. **Examples of multi-resolution image generated by `SimM`.**

tion map $\bar{\mathbf{A}}^t$ without any modification. When the $(T - T^{loc})$-th denoising step is finished, the system performs another temporal merging on all stored maps, obtaining $\bar{\mathbf{A}} \in \mathbb{R}^{W^1 \times H^1 \times N}$ that more stably indicates the source positions of generated objects:

$$\bar{\mathbf{A}} = \frac{1}{T^{loc}} \sum_{t=T-T^{loc}}^{T} \bar{\mathbf{A}}^t. \tag{4}$$

Activated Region Localization. Given the temporal-merged attention map $\bar{\mathbf{A}}$, the system locates the current activated region for each object. This is implemented by sweeping $\bar{\mathbf{A}}_k$ with a rectangular sliding window. In practice, we keep the size of the window consistent with the target box assigned by heuristic rules. And the activated region $\mathbf{b}_k^l$ in the l-th layer can be converted from the most salient window $\mathbf{b}_k^1$ found on $\bar{\mathbf{A}}_k$.

2.4. Rectify

After the $(T - T^{loc})$-th denoising step, the system starts to modify the generated cross-attention map for rectification. Note that in the following statements, $\mathbf{A}$ denotes the cross-attention maps generated before applying Softmax$(\cdot)$. Besides, the maps from the first and last cross-attention layers are not modified as we have observed that doing so improves the quality of object generation in practice.

Activation Transfer. Since the size of the localized source activated region $\mathbf{b}_k^l$ and the assigned target box $\widehat{\mathbf{b}}_k^l$ are kept the same, the activation values of the source region can be directly duplicated to the target region, while the original region is filled with minimum values. In this way, `SimM` easily realizes the movement of the object. Even if the target boxes are obtained by other means (*e.g.*, user-provided) rather than heuristic rules, this simple transfer remains valid after reshaping the source activated region.

Intra-Map Activation Enhancement and Suppression. In practice, we have found that some objects fail to appear due to the insufficient activations in the cross-attention maps. Also, one object may not be exactly in its target area even after the transfer. Therefore, for the object of the k-th token, the system continues to modify the attention map by enhancing the activations in $\widehat{\mathbf{b}}_k^l$. Meanwhile, to avoid the object appearing in non-target areas, the signal outside $\widehat{\mathbf{b}}_k^l$ is suppressed. Formally, we have

$$\mathbf{A}_k^{t,l}(i,j) \leftarrow \begin{cases} \mathbf{A}_k^{t,l}(i,j) \cdot \alpha & \text{if } (i,j) \text{ in } \widehat{\mathbf{b}}_k^l \\ \mathbf{A}_k^{t,l}(i,j) \, / \, \alpha & \text{if } (i,j) \text{ not in } \widehat{\mathbf{b}}_k^l \end{cases}, \tag{5}$$

where $l \in [2, L-1]$, and the hyperparameter $\alpha \in \mathbb{R}^+$ denotes the strength of the adjustment.

Inter-Map Activation Enhancement and Suppression. The intra-map activation adjustment further enhances the control over the position of individual objects. However, due to the lack of interference between attention maps, the overlap of activated areas on different maps can lead to conflict and confusion in the generation of multiple objects. To avoid the issue, given its corresponding attention map $\mathbf{A}_k^{t,l}$ of each object, our system generates an adjustment mask $\mathbf{M}_k^{t,l}$ for other maps:

$$\mathbf{M}_k^{t,l} = 1 - \text{Softmax}(\mathbf{A}_k^{t,l}), \tag{6}$$

where the mask adjusts the attention value of other maps:

$$\mathbf{A}_g^{t,l} \leftarrow \mathbf{M}_k^{t,l} \odot \mathbf{A}_g^{t,l}, \;\; \text{for } g \in [1, N] \text{ and } g \neq k. \tag{7}$$

In this way, after applying Softmax$(\cdot)$, the activated regions on different maps can be staggered to reduce conflicts.

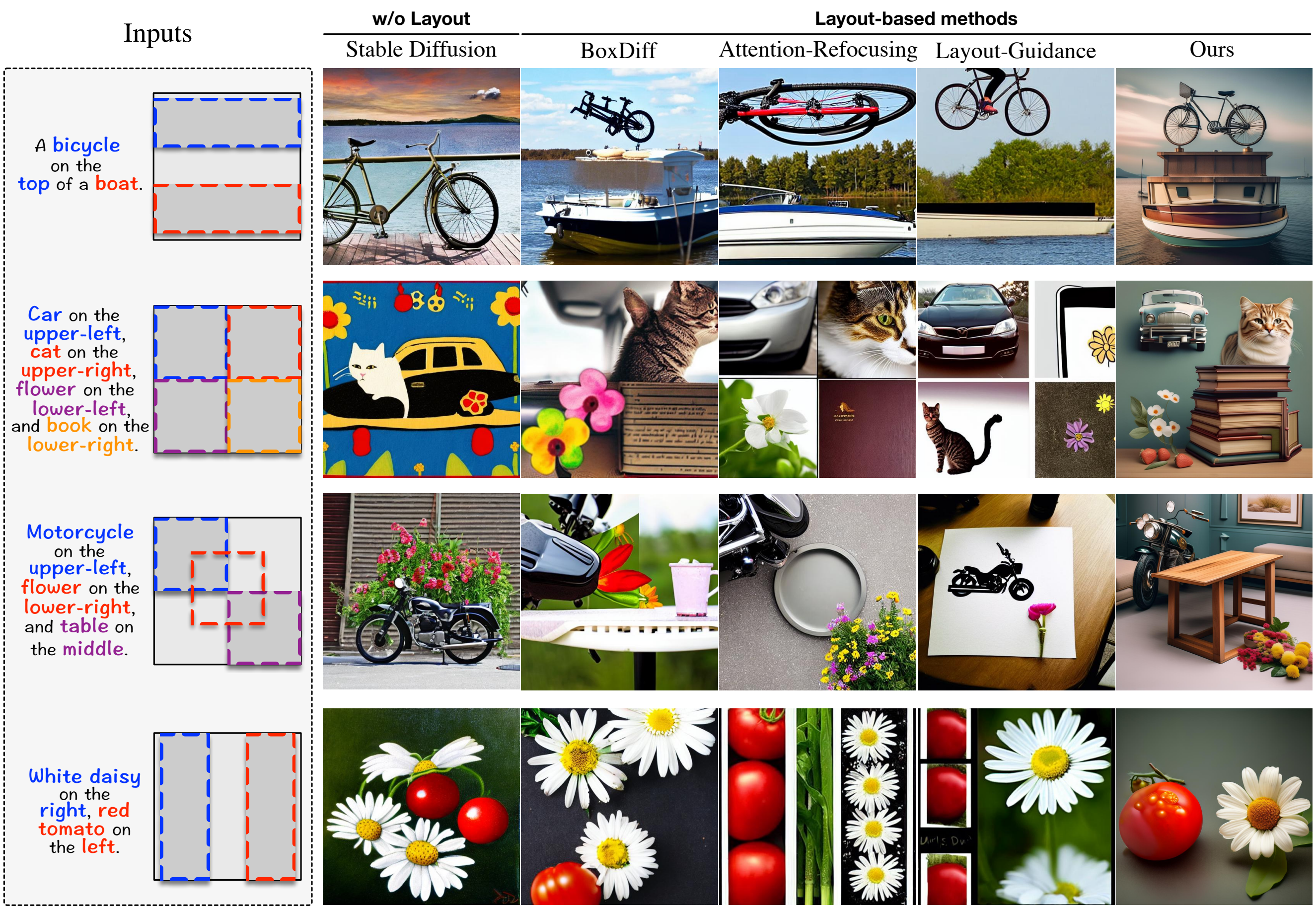

Figure 5. **Qualitative comparisons on DrawBench and SimMBench.** Textual prompts require to generate multiple objects with relative and superlative spatial relations.

3. Experiments

Datasets. We utilize different datasets to evaluate the effectiveness for both relative and superlative layout requirements. For prompts involving relative spatial relations, we use a subset of 20 prompts from the DrawBench [35] dataset, which is a common choice of previous works [25]. However, there is a lack of an appropriate dataset that addresses prompts concerning superlative spatial relations. Therefore, we present a benchmark **SimMBench** consisting of 203 prompts, where each prompt contains 1 to 4 objects, and each object has superlative layout requirements. Details are provided in *Supplementary Material*.

Baselines. We select Stable Diffusion [32], Layout-Guidance [6], Attention-Refocusing [25] and BoxDiff [40] as baselines in the main comparison. We adopt the official implement and default hyperparameters for all baselines.

Evaluation Metrics. The generation accuracy [25] is adopted as the primary evaluation metric. Specifically, a generated image will only be considered correct if all objects are correctly generated and their spatial positions or relations, color, and other possible attributes align with the corresponding phrases in the prompt. Following previous studies [40], we also report the CLIP-Score [15], which measures the similarity between the input text features and the generated image features. While this metric has been widely used to explicitly evaluate the fidelity to the text prompt, we highlight its reliability is limited, since CLIP struggles to understand spatial relationships and take them into account when scoring image-text pairs [38].

Implementation Details. We adopt the DDIM scheduler [37] with 20 denoising steps (*i.e.*, $T = 20$). And the number of localization steps T^{loc} is set to 1 as default. The ratio of classifier-free guidance is set to 5. Adjustment strength α is set to 10. Four images are randomly generated for each evaluation prompt.

3.1. Main Results

Quantitative results. Tab. 1 shows the quantitative comparison results between different baselines and our `SimM`. On the DrawBench dataset, our `SimM` achieves the highest generation accuracy and CLIP-Score, while outperforming

Table 1. **Quantitative comparisons with competing methods.** The generation accuracy (%) and CLIP-Score on DrawBench [35] and our presented SimMBench are reported.

Methods	DrawBench [35] Accuracy	DrawBench [35] CLIP-Score	SimMBench Accuracy	SimMBench CLIP-Score
Stable Diffusion [32]	12.50	0.3267	4.25	0.3012
BoxDiff [40]	30.00	0.3239	24.08	**0.3032**
Layout-Guidance [6]	36.50	0.3354	25.50	0.3020
Attention-Refocusing [25]	43.50	0.3339	50.71	0.3017
SimM (Ours)	**53.00**	**0.3423**	**65.16**	0.3001

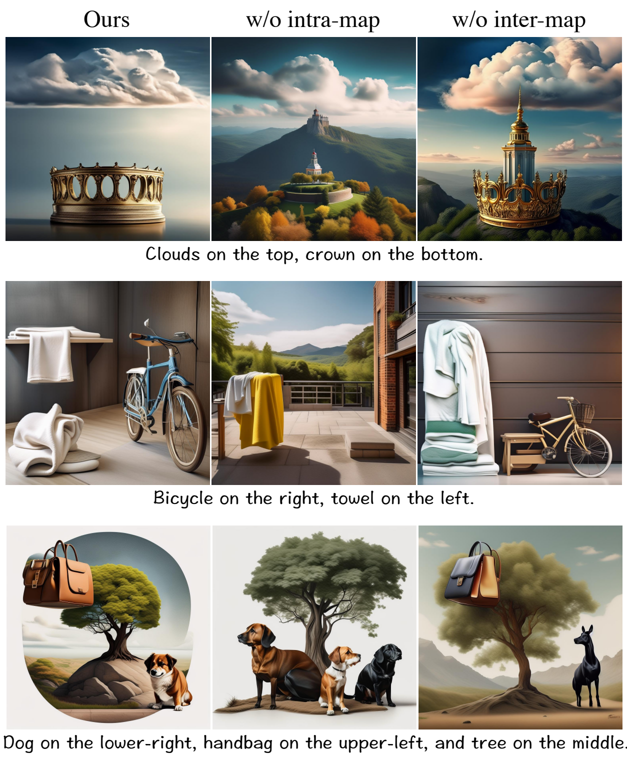

Figure 6. **Ablation study of intra-/inter-map activation adjustment.** The removal of intra-map adjustment leads to the omission of objects or positional errors, while the removal of inter-map adjustment results in fragmented or erroneous object generation.

the baselines by a significant margin of 9.5% in terms of accuracy. And on the SimMBench dataset, `SimM` not only surpasses the baselines by 14.45% in terms of accuracy but also achieves comparable CLIP-Score. The results signify the effectiveness of `SimM` system in understanding both relative and superlative relationships, leading to satisfactory rectification of layout inconsistencies.

Qualitative results. In Fig. 4, we present more multi-resolution images generated by `SimM`. Fig. 5 shows a visual comparison between the proposed `SimM` and the competing baselines. Without additional layout guidance, the images generated by the vanilla Stable Diffusion fail to convey the layout requirements specified by the textual prompt while also suffering from missing objects. The three baseline models can enhance the accuracy of the generation in

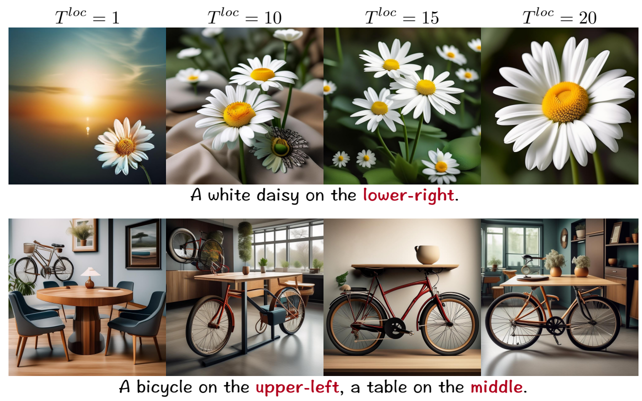

Figure 7. **Effect of the number of localization steps T^{loc}.** Initiating layout rectification at an earlier stage enhances the fidelity.

terms of layout. However, they each still suffer from respective issues. Taking the second row as an example, BoxDiff exhibits limitations in effectively controlling the layout, where the white daisies that should only appear on the right side also appear on the left and middle as well. And the images generated by Layout-Guidance and Attention-Refocusing exhibit noticeable blockiness, tearing artifacts and object deformations, which significantly degrade the quality. In contrast, our system maintains excellent image quality while rectifying the layout. We attribute this to the activation localization and movement, which allows us to preserve the generative capabilities of the base model to the maximum extent, without relying on rigid constraints imposed by loss functions.

3.2. Ablation Study

In Fig. 6, we visualize the generated images after removing the intra- and inter-map activation adjustments from `SimM`. After removing the intra-map adjustment, objects are missing (first two rows) or specified objects appear outside their target positions (the last row). This illustrates that the mechanism significantly contributes to controlling the placement of objects. Meanwhile, removing the inter-map adjustment increases the likelihood of interference from activations of other maps, which can disrupt the generation of objects in their target positions, ultimately resulting in erroneous or incomplete object generation.

3.3. Further Analysis

Effect of the number of localization steps T^{loc}. In Fig. 7, we present the visual results with layout rectification initiated at different denoising steps during the generation. It can be observed that starting the rectification from the first denoising step yields better results, ensuring that each object appears in its designated position. The later the rectification starts, the worse the correction effect, thus compromising the fidelity of the generated images. This observation is consistent with the conclusion from previous stud-

Figure 8. **Effect of adjustment strength** α**.** A value of 10 yields better layout stabilization and generation quality.

ies [3, 14], where diffusion models establish the layout in early stages and refine the appearance details in later stages.
Effect of adjustment strength α**.** We scale α from 0.1 to 50 and illustrate some generated cases in Fig. 8. Setting α to 0.1 essentially reverses the enhancement and suppression, resulting in objects appearing in non-designated positions. And setting α to 1 essentially removes the intra-map attention adjustment, leading to less effective layout rectification. Further increasing the α to 10 yields facilitates rectification and provides better control over the layout. However, excessively large values of α (*e.g.*, setting it to 50) can degrade the quality of the generated images while imposing stricter constraints on the object positions.

4. Related Work

Text-to-Image Generation. Earlier works studied text-to-image generation in the context of generative adversarial networks (GANs) [31, 39, 41, 49]. Despite their dominance, the adversarial training nature brings the issues including training instability and less diversity in generation [8]. Text-conditional auto-regressive models [9, 10, 29, 44] demonstrated more impressive results while requiring time-consuming iterative processes to achieve high-quality image sampling. Natural fitting to inductive biases of image data, the emerging diffusion models [23, 30, 32, 36] have recently demonstrated impressive generation results based on open-vocabulary text descriptions. To reduce training overhead and speed up inference, latent diffusion model [32] trims off pixel-level redundancy by applying an autoencoder to project images into latent space and generating latent-level feature maps with the diffusion process. And to align with the provided textual input, Stable Diffusion [32] further employs cross-attention mechanism to inject textual condition into the diffusion generation process.
Layout Control in Diffusion Models. Existing progress fails to fully understand the spatial relations of objects in the free-form textual descriptions and reflect them in the synthesized image, especially for complex scenes. Therefore, jointly conditioning on text and layout has been studied, where layout control signals can be bounding boxes [27, 40], segmentation maps [1, 7, 42], and key points [46]. Several methods extend the Stable Diffusion model by incorporating layout tokens into attention layers [20, 43, 47] or training layout-aware adapters [27]. However, requiring additional training on massive layout-image pairs, these approaches lack flexibility in the base model and may degrade the quality of the generated images. Therefore, recent efforts [6, 25, 40] design loss conditioned on layout constraints to update the noised latent together with denoising. Layout-Guidance [6] computes the loss by applying the energy function on the cross-attention map, Attention-Refocusing [25] constrains both cross-attention and self-attention to "refocus" on the correct regions, and BoxDiff [40] designs inner-box, outer-box, and corner spatial constraints. However, they introduce extra computational cost for gradient update, which affects the speed of generation. In contrast, our system directly modifies the activations to conform to the target for rectification, minimizing the computation overhead.
Layout Generation. Previous layout-to-image studies [6, 47] have largely neglected the discussion on layout generation and heavily relied on users to directly provide accurate layout boxes for objects. However, this necessitates assessing the legality of user input and increases the learning and interaction difficulty for users. Moreover, we have observed a substantial decline in the quality of generated images when the provided boxes are insufficiently accurate. Latest efforts [21, 25, 27] have turned to large language models like GPT-4 [24] by creating appropriate prompting templates to generate layouts, while each API request adds response time and incurs additional costs. In this paper, our system provides a light-weight solution based on dependency parsing following with heuristic rules.

5. Conclusion

In this paper, we propose a training-free layout calibration system `SimM` for text-to-image generators, which aligns the synthesized images with layout instructions in a post-remedy manner. Following a "check-locate-rectify" pipeline, `SimM` first decides whether to perform the layout rectification by checking the input prompt and the intermediate cross-attention maps. During the rectification, the system identifies and relocates the activations of mispositioned objects, where the target positions are generated by analysing the prompt with dependency parsing and heuristic rules. To comprehensively evaluate the effectiveness of `SimM`, we present a benchmark called SimMBench, which covers both simple and complex layouts described in terms of superlative relations. Through extensive qualitative and quantitative experiments, we demonstrate our superiority in improving generation fidelity and quality.

References

[1] Omri Avrahami, Thomas Hayes, Oran Gafni, Sonal Gupta, Yaniv Taigman, Devi Parikh, Dani Lischinski, Ohad Fried, and Xi Yin. SpaText: Spatio-textual representation for controllable image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18370–18380, 2023. 8

[2] Eslam Mohamed Bakr, Pengzhan Sun, Xiaoqian Shen, Faizan Farooq Khan, Li Erran Li, and Mohamed Elhoseiny. HRS-Bench: Holistic, reliable and scalable benchmark for text-to-image models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 19984–19996, 2023. 2, 3

[3] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, Tero Karras, and Ming-Yu Liu. eDiff-I: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 8

[4] Chris Bussell, Ahmed Ehab, Daniel Hartle-Ryan, and Timo Kapsalis. Generative AI for immersive experiences: Integrating text-to-image models in VR-mediated co-design workflows. pages 380–388, 2023. 2

[5] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-Excite: Attention-based semantic guidance for text-to-image diffusion models. 2023. 4

[6] Minghao Chen, Iro Laina, and Andrea Vedaldi. Training-free layout control with cross-attention guidance. *arXiv preprint arXiv:2304.03373*, 2023. 2, 4, 6, 7, 8, 1

[7] Guillaume Couairon, Marlène Careil, Matthieu Cord, Stéphane Lathuilière, and Jakob Verbeek. Zero-shot spatial layout conditioning for text-to-image diffusion models. *arXiv preprint arXiv:2306.13754*, 2023. 8

[8] Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat GANs on image synthesis. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 8780–8794, 2021. 8

[9] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, and Jie Tang. CogView: Mastering text-to-image generation via transformers. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 19822–19835, 2021. 8

[10] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-A-Scene: Scene-based text-to-image generation with human priors. In *Proceedings of the European Conference on Computer Vision*, pages 89–106, 2022. 8

[11] Songwei Ge, Taesung Park, Jun-Yan Zhu, and Jia-Bin Huang. Expressive text-to-image generation with rich text. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 7511–7522, 2023. 1

[12] Tejas Gokhale, Hamid Palangi, Besmira Nushi, Vibhav Vineet, Eric Horvitz, Ece Kamar, Chitta Baral, and Yezhou Yang. Benchmarking spatial relationships in text-to-image generation. *arXiv preprint arXiv:2212.10015*, 2022. 2

[13] Anisha Gunjal, Jihan Yin, and Erhan Bas. Detecting and preventing hallucinations in large vision language models. *arXiv preprint arXiv:2308.06394*, 2023. 2

[14] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross-attention control. In *Proceedings of the International Conference on Learning Representations*, 2023. 8

[15] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 7514–7528, 2021. 6

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proceedings of the Advances in Neural Information Processing Systems*, 2020. 2

[17] Matthew Honnibal and Mark Johnson. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, 2015. 4

[18] Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A. Smith. TIFA: accurate and interpretable text-to-image faithfulness evaluation with question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 20349–20360, 2023. 2, 3

[19] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *Proceedings of the International Conference on Learning Representations*, 2014. 2

[20] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. GLIGEN: Open-set grounded text-to-image generation. *arXiv preprint arXiv:2301.07093*, 2023. 1, 8

[21] Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. LLM-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models. *Transactions on Machine Learning Research*, 2024. 8

[22] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3D: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023. 2

[23] Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In *Proceedings of the International Conference on Machine Learning*, pages 16784–16804, 2022. 8

[24] OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 8, 2

[25] Quynh Phung, Songwei Ge, and Jia-Bin Huang. Grounded text-to-image synthesis with attention refocusing. *arXiv preprint arXiv:2306.05427*, 2023. 2, 6, 7, 8, 1

[26] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. DreamFusion: Text-to-3D using 2D diffusion. In *Proceedings of the International Conference on Learning Representations*, 2023. 2

[27] Leigang Qu, Shengqiong Wu, Hao Fei, Liqiang Nie, and Tat-Seng Chua. LayoutLLM-T2I: Eliciting layout guidance from LLM for text-to-image generation. *arXiv preprint arXiv:2308.05095*, 2023. 8, 2

[28] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning*, pages 8748–8763, 2021. 3

[29] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *Proceedings of the International Conference on Machine Learning*, pages 8821–8831, 2021. 1, 8

[30] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *arXiv preprint arXiv:2204.06125*, 2022. 8

[31] Scott E. Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *Proceedings of the International Conference on Machine Learning*, pages 1060–1069, 2016. 1, 8

[32] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10674–10685, 2022. 2, 6, 7, 8, 1

[33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. pages 234–241, 2015. 3

[34] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. DreamBooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023. 2

[35] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 36479–36494, 2022. 2, 4, 6, 7

[36] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 36479–36494, 2022. 8

[37] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *Proceedings of the International Conference on Learning Representations*, 2021. 6

[38] Sanjay Subramanian, William Merrill, Trevor Darrell, Matt Gardner, Sameer Singh, and Anna Rohrbach. ReCLIP: A strong zero-shot baseline for referring expression comprehension. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 5198–5215, 2022. 6

[39] Ming Tao, Hao Tang, Fei Wu, Xiaoyuan Jing, Bing-Kun Bao, and Changsheng Xu. DF-GAN: A simple and effective baseline for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16494–16504, 2022. 8

[40] Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. BoxDiff: Text-to-image synthesis with training-free box-constrained diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2023. 6, 7, 8, 1

[41] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1316–1324, 2018. 8

[42] Han Xue, Zhiwu Huang, Qianru Sun, Li Song, and Wenjun Zhang. Freestyle layout-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 8

[43] Zhengyuan Yang, Jianfeng Wang, Zhe Gan, Linjie Li, Kevin Lin, Chenfei Wu, Nan Duan, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. ReCo: Region-controlled text-to-image generation. *arXiv preprint arXiv:2211.15518*, 2022. 2, 8

[44] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. Scaling autoregressive models for content-rich text-to-image generation. *Transactions on Machine Learning Research*, 2022. 8

[45] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2023. 2

[46] Zhiyuan Zhang, Zhitong Huang, and Jing Liao. Continuous layout editing of single images with diffusion models. *arXiv preprint arXiv:2306.13078*, 2023. 8

[47] Guangcong Zheng, Xianpan Zhou, Xuewei Li, Zhongang Qi, Ying Shan, and Xi Li. Layoutdiffusion: Controllable diffusion model for layout-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22490–22499, 2023. 2, 4, 8, 3

[48] Yiyang Zhou, Chenhang Cui, Jaehong Yoon, Linjun Zhang, Zhun Deng, Chelsea Finn, Mohit Bansal, and Huaxiu Yao. Analyzing and mitigating object hallucination in large vision-language models. *arXiv preprint arXiv:2310.00754*, 2023. 2

[49] Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. DM-GAN: Dynamic memory generative adversarial networks for

text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5802–5810, 2019. 8